3. Ben

Program Name: Ben.java **Input File: ben.dat**

Ben was recently creating his account for a new game that he wanted to play with his friends and went to enter in his username. After successfully creating his account and joining his friends' lobby, one of his friends noticed that he spelled his username incorrectly! Ben had accidentally spelled "pigeon" as "pidgeon" as he thought that "pidgeon" sounded more phonetically correct than "pigeon."

Being the good friends that they are, Ben's friends decided to poke a little fun at him by changing their usernames to other misspellings of pigeon. More specifically, they wanted to create usernames which contain multiple noncontiguous substrings of the string "pidgeon". Help Ben determine just how good of friends he has by calculating the number of unique noncontiguous substrings there exist of the string "pigeon" in the various different usernames his friends came up with. Given that these numbers may become decently large, determine this number modulo $10^9 + 7$.

Input: The first line of input will consist of a single integer n ($1 \leq n \leq 50$) denoting the number of testcases to follow. The next n testcases will consist of a single line which consists of the username (u) for the current testcase. It is guaranteed that u consists of only the letters 'p', 'i', 'd', 'g', 'e', 'o', and 'n', although, not necessarily in that order. Additionally, it should be noted that the length of the string will fall in the range $1 \leq |u| \leq 10^3$.

Output: For each of Ben's n usernames, print out the number of non-contiguous substrings of the username that are equivalent to the string "pidgeon" modulo $10^9 + 7$.

Sample input:

```
4
pidgeon
pidgeonpidgeon
pigeon
ppiiddggeeoonnppiiddggeeoonn
```

Sample output:

```
1
8
0
1024
```